一种优化的硬阈值追踪算法的研究 *

陈　暄[1]，潘春平[1]，龙　丹[2]
(1. 浙江工业职业技术学院, 浙江 绍兴 312000; 2. 浙江大学, 杭州 310058)

摘　要：硬阈值追踪算法是压缩感知中一种重要的重构算法，该算法的本质是一个最小二乘问题，存在复杂度高，收敛性差、运行时间长等缺点。引入 Nesterov 方法用以优化稀疏解的凸松弛现象，引入逐次松弛迭代法优化传统硬阈值线性方程组，理论证明优化结果具有良好的收敛性，仿真实验说明优化后的算法有效降低算法复杂度，降低了运行时间。

关键词：硬阈值；追踪；Nesterov；逐次超松弛迭代法
中图分类号：TP391　　doi: 10.19734/j.issn.1001-3695.2018.10.0769

Research on optimization hard threshold tracking algorithm

Chen Xuan[1], Pan Chun Pin[1], Long Dan[2]
(1. *Zhejiang Industry Polytechnic College*, *Shaoxing Zhejiang* 312000, *China*; 2. *Zhejiang University*, *Hangzhou* 310058, *China*)

Abstract: The hard threshold tracking algorithm is an important reconstruction algorithm in compressed sensing, which is essentially a least squares problem and has the disadvantages of high complexity, poor convergence and long running time. This paper introduced the Nesterov method to optimize the convex relaxation phenomenon of sparse solutions, and introduced the successive relaxation iterative method to optimize the traditional hard threshold linear equations. The theory proves that the optimization results have good convergence. Simulation experiments show that the optimized algorithm effectively reduces the complexity of the algorithm and the running time.

Key words: hard threshold; tracking; nesterov ; successive over relaxation method

0　引言

压缩感知理论[1~5]突破传统的 Nyquist 定理的严格限制,实现了对信号的采样能够在较低的采样率下进行,并同时完成对信号的压缩,在重构一端通过非线性化的方式进行信号的重构。压缩感知理论的提出使得信号在采样过程中不再主要依赖信号的宽度,这样能够有效的降低了信号传输和存储过程中所需要的代价。因此研究如何能够高效的恢复原始信号成为了重构算法的主要方向,目前在压缩感知中的重构算法有凸松弛算法,贪婪追踪算法和组合优化算法。凸松弛算法具有观测次数少,复杂度偏高的特点,其代表有基追踪算法[6](basic pursuit, BP),梯度投影稀疏重建算法[7](gradient projection for sparse, GPSR),最小角度回归法[8](least angle regreesion, LARS);贪婪追踪算法具有算法具有节点,速度快,复杂度低的特点,其代表有正交匹配追踪算法[9](orthogonal matching pursuit, OMP),正则化正交匹配算法[10](regularized orthogonal matching pursuit, ROMP),分段正交匹配追踪算法[11](stagewise orthogonal matching pursuit, STOMP)等;组合优化算法主要是稀疏傅里叶变换算法[12](sparse Fourier transform, SFFT)。迭代硬阈值[13](iterative hard thresholding, IHT)是一种凸松弛算法,优点是适合大规模高维数据处理和实时数据处理,缺点是测量矩阵在有限等距特性条件下需要满足最大奇异值小于 1 的限制条件,因此导致了算法对对测量矩阵的尺度要求高,运算运行时间长,复杂度高。学者从不同的角度其进行了研究,取得了一些成果。文献[14]提出 S-闭包的分块相干性和约束等距特性(RIP)概念;根据冗余字典分块相干性的特点,对更新支撑集部分对算法进行改进,仿真实验说明有效的降低了算法的收敛性;文献[15]提出一种基于 SHTP 算法的联合重构算法来求解分布式压缩感知问题,仿真结果表明该算法具有较大的信号重构噪声比和较小的平均支撑势误差,可实现信号值的精确重构;文献[16]提出一种压缩采样硬阈值的重构算法,该算法在去除原子时结合硬阈值思想,保证了每次迭代更加的精确,仿真实验说明算法具有更好的重构性和抗躁能力;文献[17]提出了一种新的部分硬阈值(PHT)算子,它类似于硬阈值算子,限制了支持集在一次迭代中可以改变的量,仿真实验说明该算法提高算法效率;文献[18]提出了一种扩展的稀疏恢复理论并通过在硬阈值算法中使用取得了较好的效果,能够有效的恢复信号;文献[19]提出了一种复杂的洛伦兹迭代硬阈值算法,仿真实验说明该算法能够有效的恢复信号,具有较好的效果;文献[20]将硬阈值算法用于从少量线性测量中最优地恢复稀疏信号,然后对 CS 进行去噪和传统去噪处理的比较研究,该方法能够有效的降低噪声对信号的影响;文献[21]提出在硬阈值算法中将黎曼优化算法用于低秩矩阵恢复的思想,保证共轭梯度算法线性收敛到基础秩矩阵。 经验评估表明,算法能够从几乎所需的最小测量数量中恢复低秩矩阵;文献[22]提出了一种基于广义期望最大化(GEM)噪声稀疏重构的降噪硬阈值方法,取得了较好的效果。

收稿日期：2018-10-17；**修回日期：**2018-12-07　**基金项目：**国家自然科学基金资助项目（LQ18A010003，11426205）；绍兴市科技局资助项目（2015B70013）

作者简介：陈暄（1979-），男，江西南昌人，副教授，硕士，主要研究方向为云计算、无线传感；潘春平（1978-），男，湖南人，副教授，硕士，主要研究方向为算法设计；龙丹（1975-），男，湖南人，讲师，博士，主要研究方向为图像处理和分析、算法设计.

本文在文献[23]关于硬阈值算法研究的基础上,引入Nesterov和逐次超松弛替代法来进行改进,并证明了改进后的算法具有良好的收敛性,在仿真实验的指标测评中取得了较好的效果。

1　硬阈值追踪算法

所有的压缩重构算法都必须遵守限制自同构性质(RIP)[4]。设 $A \in R^{M \times N}$ 的 s 阶的自同构系数 $\delta_s = \delta_s(A)$ 定义为满足式(1)的最小的 δ。

$$(1-\delta)\|x\|_2^2 \le \|Ax\|_2^2 \le (1+\delta)\|x\|_2^2 \tag{1}$$

对于任意的 s 稀疏的向量 $x \in R^N$。文献[20,21]证明了当 A 满足了 RIP 条件后,这些算法都是收敛的。

硬阈值追踪算法结合了 IHT 的硬阈值策略和 CoSaMP 的投影算法,迭代公式如下:

$$S_{n+1} = L_s(x_n + A^*(y - Ax_n)) \tag{2}$$

$$x_{n+1} = \arg\min_z \|y - Az\|_2, supp(z) \subseteq S_{n+1} \tag{3}$$

式(3)中的 $supp(x^n)$ 表示 x^n 的支撑集合。$L_s(x)$ 表示 x 绝对值中最大的 s 个元素的支撑。硬阈值的迭代步骤(即式(3))的本质形式是一个最小二乘问题。随着压缩感知中的信号重构的问题规模逐渐增大,采用直接解决最小二乘的问题就会非常的耗时,因此,文献[20]中提出的一种快速硬阈值的算法能够有效的解决大规模下的稀疏恢复的问题,它采用的梯度法来解决迭代中的最小二乘问题,当使用梯度法的时候,取步长为 1,得到:

$$\|x_{n+1} - x\|_2 \le \sqrt{\frac{2\delta_{3s}^2 + \delta_{3s}^{2k+2}(1-3\delta_{3s}^2)}{1-\delta_{2s}^2}}\|x_n - x\|_2 \tag{4}$$

其中:x 是任意 s 稀疏的满足 $Ax = y$ 的向量,$\{x_n\}_{n=0,1,2...}$ 是快速硬阈值的算法产生的迭代序列。从上面的研究中发现,式(3)一方面是一个待优化问题,采用优化效果更好的 Nesterov 可以提高优化的效果,从另一个方面来看式(3)也是一个正定的线性方程组,需要使用线性解决办法更好的逐次松弛替代法解决问题,本文阐述并证明了式(3)通过以上两种算法的改进得到的算法仍然是收敛的。

2　改进的硬阈值追踪算法

2.1　Nesterov 方法

该方法是一种用以求解加速求解光滑优化问题的算法,其表达式为

$$\min f(x) \tag{5}$$

其中: f 是 Lipschitz 梯度 $L = L(f)$,即满足

$$\|\nabla f(x_1) - \nabla f(x_2)\|_2 \le L\|x_1 - x_2\|_2, x_1, x_2 \in dom(f) \tag{6}$$

算法步骤如下所示。

算法 1: Nesterov 算法

1: 初始化: $y^0 = x^0, \theta_0 = 1$

2: $while$ "$not\ converage$" $,do$

$x^{k+1} = y^k - h_k \nabla f(y^k)$

$\beta_{k+1} = \frac{1}{2}(1-\theta_k)(\sqrt{\theta_k^2 + 4} - \theta_k)$

$y^{k+1} = x^k + \beta_k(x^{k+1} - x^k)$

$\theta_{k+1} = \frac{1}{2}\theta_k(\sqrt{\theta_k^2 + 4} - \theta_k)$

$k = k+1$

$endwhile$

3: 输出最终的迭代结果 x^k

算法中的 h_k 是步长参数,设定 $h_k = \frac{1}{L}$,文献[12]证明存在如下的关系

$$f^* - f(y^k) \ge \frac{2L\|y^0 - y^*\|_2^2}{(k+1)^2} \tag{7}$$

其中的 y^* 就是问题(5)的解且满足 $f^* = f(y^*)$。当式(3)使用算法 1 求解的时候,目标函数为 $f(x) = \|y - A_n x\|_2^2$, $A_n = A(:, S_n) \in R^{M \times s}$,根据 A 具有限制自同构性质得到:

$$\|A_n x\|_2^2 \le (1+\delta_s)\|x\|_2^2 < 2, \forall x \in R^s \tag{8}$$

因此,从式(8)中发现 $\|A_n x\|_2^2 < 2$,即 $\|A_n x\|_2 < 2$,因此有

$$\|\nabla f(x_1) - \nabla f(x_2)\|_2 = \|A_n^* A_n(x_1 - x_2)\|_2 < 2\|x_1 - x_2\|_2 \tag{9}$$

2.2　逐次超松弛迭代法

式(3)其本质也是求解下列线性方程

$$A_n^* A_n x = A_n^* y \tag{10}$$

$A_n = A(:, S_n) \in R^{M \times s}$,因为 A 具有限制自同构性质,所以,$A_n^* A_n$ 是正定矩阵,把矩阵 $A_n^* A_n$ 分裂成 $A_n^* A_n = D - (D - A_n^* A_n) = D - L - L^*$,其中,$D = diag(d_1, d_2, \cdots d_n)$,$d_i > 0$ 是对称正定矩阵 $A^* A$ 的对角部分,$-L$ 是 $A^* A$ 的严格下三角部分。令 $A^* A = M - N$

$$M = \frac{1}{w}(D - wL)$$

$$N = \frac{1}{w}((1-w)D + wL^*), w \ne 0$$

则得到逐次超松弛迭代法

$$x^{(k)} = wD^{-1}(Lx^{(k)} + L^* x^{(k-1)} + A^* y) + (1-w)x^{(k-1)} \tag{11}$$

即

$$x^{(k)} = (D - WL)^{-1}((1-w)D + WL^*)x^{(k-1)} + w(D - WL)^{-1} A_n^* y \tag{12}$$

其中: w 是松弛因子,因此迭代法式(12)的迭代矩阵 $G_w = (D - wL)^{-1}((1-w)D + wL^*)$,上述迭代需要显示计算 $A^* A$,若在第 k 步,令 $Z^{(1)} = x^{(k-1)}$, $r^{(1)} = b - Ax^{(k-1)}$,则可得不需要显示计算 $A^* A$。算法步骤如下所示。

算法 2　逐次超松弛迭代法算法

1: $Z^{(1)} = x^{(k-1)}$, $r^{(1)} = A_n^* y - A_n x^{(k-1)}$

2: $Z^{(j+1)} = Z^{(j)} + \delta_j e_j$

$r^{(j+1)} = r^{(j)} - \delta_j a_j$

$\delta_j = w a_j^* r^{(j)} / d_j, j = 1, \cdots n$

$x^{(k)} = Z^{(n+1)}$

3: 输出迭代结果 x^k

2.3　本文算法的收敛性分析

本文算法步骤如下所示。

算法 3　本文算法

1: 初始化: x^0

2: $while$ "$not\ converage$" $,do$

$S_{n+1} = L_s(x_n + A^*(y - Ax_n))$

$x_{n+1}^0 = (x_n + A^*(y - Ax_n))S_{n+1}$

$x_{n+1}^{(k)} = (D - WL)^{-1}((1-W)D + WL^*)x_{n+1}^{k-1} + w(0 - WL)^{-1} A_n^* y$　　$x_{n+1} = x_{n+1}^{(k)}$

$k = k+1$

$if\ \|x_{n+1} - x_n\|_2 < \varepsilon$　转步骤 3,否则转步骤 2

$endwhile$

3: 输出结果 x^k

下面的过程就来证明算法 3 是具有收敛性的

定理 1　设 $A_n \in R^{M \times S}, x \in R^S, y \in R^M$, A_n^* 是 A_n 的共轭转置矩阵,则当 $0 < W < 2$ 的时候,迭代方法 3 是收敛的。

证明　设 λ 是逐次超松弛迭代法方法的迭代矩阵 G_w 的任意一个特征值,$X \in R^S$ 为其相对应的特征向量,则有等式 $((1-w)D + wL^*)x = \lambda(D - WL)x$,因此

$(1-W)x^*Dx+Wx^*L^*x=\lambda(x^*Dx-Wx^*Lx)$，因为 $A_n^*A_n$ 对称正定

所以 $x^*Dx=q>0$

$$x^*Lx=\alpha+\beta i$$
$$x^*L^*x=(x^*Lx)^*=\alpha-\beta i$$
$$x^*A_n^*A_nx=x^*(D-L-L^*)x=q-2\alpha$$

其中，$\alpha,\beta\in R$

于是，$\lambda=\dfrac{q-wq+w\alpha-w\beta i}{q-w\alpha-w\beta i}$

可得 $|\lambda|^2=\dfrac{[q-w(q-\alpha)]^2+w^2\beta^2}{(q-w\alpha)^2+w^2}$

当 $0<w<2$ 的时候，有

$[q-w(q-\alpha)]^2-(q-w\alpha)^2=qw(2-w)(2\alpha-q)<0$ 因此，$|\lambda|<1$

即有 $\rho(Gw)<1$

因此，迭代方法是收敛的

因此接下来证明算法 3 具有的收敛性

引理 1 根据文献[20]中的假设 A 具有限制自同构性质，U 是指标集。因此对于任意的向量 v 来说，存在

$$\|((I-A^*A)v)_U\|_2\le\delta_t\|v\|_2 \quad (13)$$

其中，$|U\cup supp(v)|\le t$

引理 2 在算法 3 中，定义 $\lim_{l\to\infty}x_n^l=x_n^*$。因此对于任意的 s 稀疏满足 $supp(z)\subseteq S_n$，向量 z 和任意 s 稀疏的满足 $y=Ax$ 的向量 x，有

$$\langle x-x_n^*,z\rangle\le\delta_{2s}\|x-x_n^*\|_2\|z\|_2 \quad (14)$$

证明 由于 Ax_n^* 是 $\{Az,supp(z)\subseteq S_n\}$ 上的 y 近似最佳，因此有

$$\langle y-Ax_n^*,Az\rangle=0,if\ \ supp(z)\subseteq S_n \quad (15)$$

因为 $y=Ax$，式(15)重写为

$$\langle Ax-x_n^*,Az\rangle=0,if\ \ supp(z)\subseteq S_n \quad (16)$$

将式(16)和引理 1 代入式(15)，定义 $U_n=supp(x-x_n)$

$\langle x-x_n^*,z\rangle=\langle x-x_n^*,(I-A_n^*A)z\rangle$ =

$\langle x-x_n^*,[(I-A_n^*A)z]_{U_n}\rangle\le\delta_{2s}\|x-x_n^*\|_2\|z\|_2$

引理 3 在算法 3 中，对于任意稀疏 s 的满足 $y=Ax$ 的向量 x，$\{x_n\}_{n=1,2,3,\dots}$ 满足

$$\|(x_{n+1}-x)_{S_{n+1}}\|_2\le\delta_{2s}\|x_n-x\|_2 \quad (17)$$

证明 设 $S=supp(x)$，得到 $\|(x_n+A^*(y-Ax_n))_S\|_2^2\le\|(x_n+A^*(y-Ax_n))_{S_{n+1}}\|_2^2$，

在两边同时减去 $\|(x_n+A^*(y-Ax_n))_{S_n,S_{n+1}}\|_2^2$，得到

$$\|(x_n+A^*(y-Ax_n))_{S_n,S_{n+1}}\|_2$$
$$\le\|((I-A^*A)(x_n-x)_{S_n,S_{n+1}}\|_2$$
$$\ge\|(x-x_{n+1})_{S_{n+1}}\|_2-\|(I-A^*A)(x_n-x))_{S_n,S_{n+1}}\|_2$$ 因此，得到
$$\|(x-x_{n+1})_{S_{n+1}}|_2\le\|((I-A^*A)(x_n-x))_{S_n,S_{n+1}}\|_2$$

引理 4 设 $x_{n+1}^{(k)}$ 是由逐次迭代方法产生的序列，则 $\|x_n^{(k)}-x_n^*\|\le\dfrac{\|G_w\|^k}{1-\|G_w\|_2}\|x_n^{(0)}-x_n^*\|$

定理 2 设定算法 3 中产生的序列 x_{n+1}^i 满足 $\|x_n^i-x_n^*\|\le\gamma(i)\|x_n^0-x_n^*\|$，因此，对于任意 s 稀疏的满足 $y=Ax$ 的向量 x，因此算法 3 中的产生的 x_n 满足

$$\|x_n-x\|_2\le\rho^n\|x_0-x\|_2 \quad (18)$$

$$\rho=(1+\delta_s)(\delta_{2s}+\gamma(l))\sqrt{\frac{1+\delta_{2s}}{1-\delta_{2s}}}+\delta_{2s}\gamma(l)+\delta_{2s}$$

证明 本文可以得到

$$\|(x_{n+1}-x)_{S_{n+1}}\|_2^2=\langle x_{n+1}-x,(x_{n+1}-x)_{S_{n+1}}\rangle=$$
$$\langle x_{n+1}-x_n^*,(x_{n+1}-x_n)_{S_{n+1}}\rangle+\langle x_{n+1}^*-x,(x_{n+1}-x_n)_{S_{n+1}}\rangle$$

注意到 $x_{n+1}=x_{n+1}^l$，因此，

$$x_{n+1}-x_n^*,(x_{n+1}-x_n)_{S_{n+1}}<\gamma(l)\|x_{n+1}^0-x_{n+1}^*\|_2\times\|(x_{n+1}-x_n)_{S_{n+1}}\|_2$$

通过引理 4

$$x_{n+1}^*-x,(x_{n+1}-x_n)_{S_{n+1}}<\delta_{2s}\|x_{n+1}^*-x\|_2\times\|(x_{n+1}-x_n)_{S_{n+1}}\|_2$$

化简得到

$$\|(x_{n+1}-x)_{S_{n+1}}\|_2$$
$$\le\delta_{2s}\|x_{n+1}^*-x\|_2+\gamma(l)\|x_{n+1}^0-x_{n+1}^*\|_2$$
$$\le(\delta_{2s}+\gamma(l))\|x_{n+1}^*-x\|_2+\gamma(l)\|x_{n+1}^0-x\|_2 \quad (19)$$

根据 A 的限制自同构性质，可以得到

$$\|x_{n+1}^*-x\|_2^2\le\frac{1}{1-\delta_{2s}}\|Ax_{n+1}^*-y\|_2^2$$

$$\le\frac{1}{1-\delta_{2s}}\|(D-WL)^{-1}((1-W)D+WL^*)x_{n+1}^{k-1}+w(0-WL)^{-1}A_n^*y-y\|_2^2\le\frac{1+\delta_{2s}}{1-\delta_{2s}}\|(x_n)_{S_{n+1}}-x\|_2^2$$

$$=\frac{1+\delta_{2s}}{1-\delta_{2s}}\|((D-WL)^{-1}((1-W)D+WL^*)x_{n+1}^{k-1}+w(0-WL)^{-1}A_n^*y-x)_{S_{n+1}}\|_2^2+\frac{1+\delta_{2s}}{1-\delta_{2s}}\|(x)_{S_{n+1}}\|_2^2$$

$$\le\frac{1+\delta_{2s}}{1-\delta_{2s}}\|(x_n-x)\|_2^2+\frac{1+\delta_{2s}}{1-\delta_{2s}}\|(x_{n+1}-x)_{S_{n+1}}\|_2^2 \quad (20)$$

由算法 3 的格式，有 $x_{n+1}^0=(x_n+A^*(y-Ax_n))_{S_{n+1}}$，根据引理 1 得到

$$\|x_{n+1}^0-x\|_2^2=\|(x_n+A^*(y-Ax_n))_{S_{n+1}}-x\|_2^2$$
$$=\|(x_n-x+A^*(y-Ax_n))_{S_{n+1}}\|_2^2+\|x_{S_{n+1}}\|_2^2=\|[(I-A^*A)(x-x_n)]_{S_{n+1}}\|_2^2+\|x_{S_{n+1}}\|_2^2$$
$$\le\delta_{2s}^2\|x-x_n\|_2^2+\|(x_{n+1}-x)_{S_{n+1}}\|_2^2$$

两边开方得到

$$\|x_{n+1}^0-x\|_2\le\delta_{2s}\|x-x_n\|_2+\|(x_{n+1}-x)_{S_{n+1}}\|_2$$

因此，将该结果和式(20)代入式(19)得到

$$\|(x_{n+1}-x)_{S_{n+1}}\|_2$$
$$\le[(\delta_s+\gamma(l))\sqrt{\frac{1+\delta_s}{1-\delta_s}}+\delta_{2s}\gamma(l)]\|x_n-x\|_2+$$
$$[(\delta_s+\gamma(l))\sqrt{\frac{1+\delta_s}{1-\delta_s}}+\delta_{2s}\gamma(l)]\|(x_{n+1}-x)_{S_{n+1}}\|_2$$

由引理 3 得到 将式(18)代入上式中得到

$$\|(x_{n+1}-x)_{S_{n+1}}\|_2$$
$$\le[(1+\delta_{2s})(\delta_{2s}+\gamma(l))\sqrt{\frac{1+\delta_{2s}}{1-\delta_{2s}}}+\delta_{2s}\gamma(l)]\|x_n-x\|_2$$

最后可以得到

$$\|x_{n+1}-x\|_2\le\|(x_{n+1}-x)_{S_{n+1}}\|_2+\|(x_{n+1}-x)_{S_n}\|_2\le\rho\|x_n-x\|_2$$

因此得到：$\|x_{n+1}-x\|_2\le\rho\|x_n-x\|_2$，因此本文算法证明成立。

3　实验仿真

3.1　算法重构性能对比

为了进一步说明本文算法重构性能，选择信号长度 N 为 256，稀疏度为 K 的随机信号 x，设定 Φ 为 $M\times N$ 为随机高斯矩阵，测量值为 $y=\Phi x$，实验信号的稀疏度比值设定为[0.05,0.3]之间，重构精度以误差率 $f=\|x^{'}-x\|_2^2/\|x\|_2^2$ 作为判定比较标准，设定当误差率小于 1e-004 的时候，说明信号重构成功，当误差率大于等于 1，则说明信号重构失败，将 OMP,SP,CoSamp,IHT 与本文算法在重构性能和运算复杂度两个指标上进行对比。设定实验运行 300 次，对这 5 种算法的实验数据结果按照取平均值进行计算。图 1 和 2 显示了这 5 种算法在不同的稀疏度下的重构精度以及算法所需要的时间。

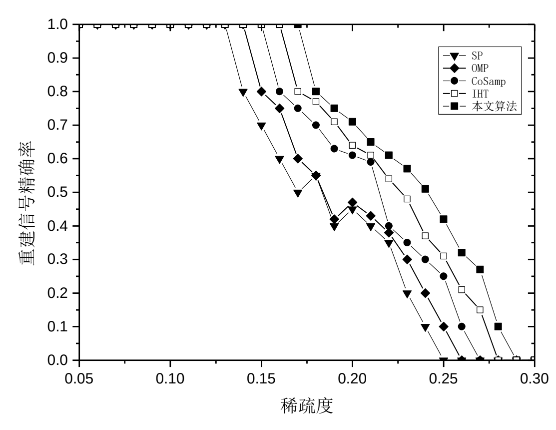


图 1　不同算法下的重构精确率对比

Fig. 1　Comparison of the Reconstructing Accuracy under Different Algorithms

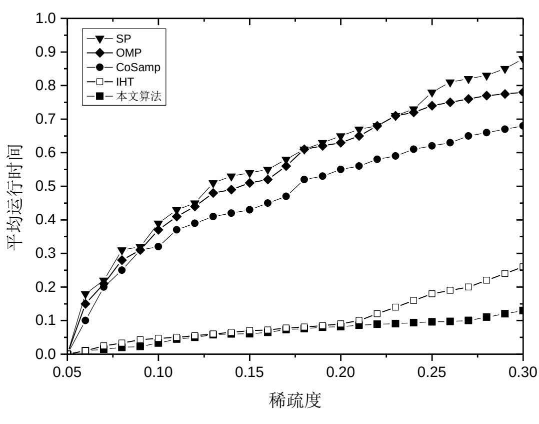


图 2　不同算法下的平均运行时间

Fig. 2　Average Operation Time under Different Algorithms

图 1 显示了在不同的稀疏度数值下的 5 种算法重构精度的对比,从图中发现随着稀疏度数值不断增大,5 种算法的重构精度都在不断的减少,从图中的曲线中发现本文算法是在稀疏度值接近 0.2 位置才开始下降,是所有曲线下降最晚的,这说明本文算法在重构精度方面具有一定优势。图 2 显示了各种算法平均运行时间,伴随着 K 的不断增大,本文算法的运行时间相比于其他算法是最短的。

3.2　算法抗躁性能对比

为了进一步说明该算法具有的抗躁效果,本文选取了选取稀疏度 K 值为 30 的上述信号,设定 e 为干扰噪声,观测向量表达式为 $y=\Phi x+e$,干扰噪声设定分别为 0.01,0.05,0.1,0.15,分别对 OMP,SP,CoSamp,IHT 以及本文算法进行抗噪性对比,实验结果如表 1 所示。

表 1　算法抗躁性性能比较

Table 1　Comparison of the Algorithms' Anti-manic performance

算法	e=0.01	e=0.05	e=0.1	e=0.15
OMP	0.015	0.25	0.62	0.97
SP	0.017	0.1	0.22	0.65
CoSamp	0.019	0.15	0.26	0.46
IHT	0.013	0.09	0.21	0.29
本文算法	0.011	0.06	0.16	0.19

从表 1 中发现本文算法的抗躁效果要优于 OMP,SP 等传统的重构算法,特别是随着干扰噪声数值逐渐增大,本文算法的噪声数值增长最小,说明具有良好的抗躁性。

3.3　图像对比

对图像进行实验,选取256×256的Lena图像,采用高斯矩阵为观测矩阵,稀疏基选择小波变换矩阵,设置观测次数为[100,200],进行 500 次实验后的平均数据。表 2 显示了 5 种算法的重构图像的 PSNR 值,从表 2 中可以发现,本文算法的重构图像的 PSNR 值相比 CoSamp 和 IHT 提高了 0.2-0.6dB,比 OMP 和 SP 提高了 0.8~0.9 dB.

表 2　不同算法下的重构性能和时间对比

Table 2　Comparison of Reconstructing Performance and Time under Different Algorithms

算法	M=100		M=200	
	PSNR	时间	PSNR	时间
OMP	28.94	6.99	31.07	9.76
SP	29.13	7.73	31.23	21.58
CoSaMP	29.71	16.52	31.29	42.95
IHT	29.82	4.48	31.24	4.57
本文算法	29.98	16.75	31.86	44.07

为了测试该算法在确定型观测矩阵下精够重构图像,观测矩阵部分选取部分哈达玛矩阵,实验结果如图 3 所示。

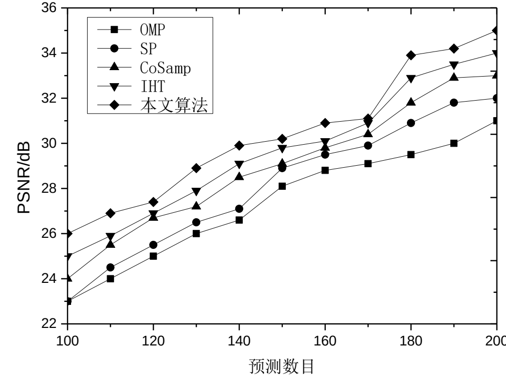


图 3　不同算法下的哈达玛矩阵下的 PSNR 值对比

Fig. 3　Comparison of PSNR Values under Hadamard Matrix under Different Algorithms

图 3 中显示了随着观测数目不断增加,五种算法的重构图像的PSNR数值也不断提高,从整体上看本文算法均好于其他几种算法,采用部分哈达玛矩阵也能精确地重构图像。因此本文算法能够有效地重构信号。

4　结束语

重构算法一直都是压缩感知中的重要研究方向,本文针对硬阈值追踪算法自身存在的不足,提出了引入 Nesterov 和逐次超松弛进行改进，仿真实验说明本文算法能够有效的提高收敛性,降低运行时间,提高了重构效果。

参考文献：

[1] Donoho D L. Compressed sensing [J]. IEEE Trans on Information Theroy, 2006, 52 (4): 1289-1306.

[2] Baraniuk R. Compressed sensing [J]. IEEE Signal Processing Magazine, 2007, 24 (4): 118-121.

[3] Candès E J, Romberg J, Tao T. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information [J]. IEEE Trans on Information Theroy, 2006, 52 (2): 489-509.

[4] Candès E J, Tao T. Decoding by linear programming [J]. IEEE Trans on Information Theroy, 2005, 51 (12): 4203-4215.

[5] Bruckstein A M, Donoho D L, Elad M. From sparse solutions of systems of equations to sparse modeling of Signals and images [J]. SIAM Review, 2009, 51 (1): 34-81.

[6] Chen Shaobing S, Donoho D L, Saunders M A.. Atomic decomposition by basis pursuit [J]. SIAM Journal of Science Computing, 1988, 20 (1):

33-61.

[7] Kim S J, Koh K, Lustig M. An interior-point method for large-scale L_1-regularized least squares [J].IEEE Journal of Selected Topics in Signal Processing, 207, 1(4): 606-617.

[8] Mehrotra S. On the implementation of a primal-dual interior point method [J].Society for Industrial and Applied Mathematics, 1992, 2(4): 575-601

[9] Tropp J A. Gilbert A C. Signal recovery from random measurements via orthogonal matching pursuit [J]. IEEE Trans on Information Theroy, 2007, 53 (12): 4655-4666

[10] Needell D,Vershynin R. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit [J]. IEEE Journal of Selected Topics in Signal Processing, 2010, 4 (2): 310-316

[11] Donoho D L, Tsagi Y, Drori I, *et al.* Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit [J]. IEEE Trans on Information Theroy, 2012, 58 (2): 1094-1121

[12] Hassanieh H, Indyk P, Katabi D,*et al.* Nearly optimal sparse fourier transform [C]// Proc of Annual ACM Symposium on Theory of Computing.New York:ACM Press,2012:563-578.

[13] Blumensath T, Davies M E. Iterative hard thresholding for compressed sensing [J]. Applied and Computational Harmonic Analysis, 2009, 27, 265-274.

[14] 陈鹏，孟晨，王成．基于冗余字典的联合稀疏同步迭代硬阈值算法 [J]. 计算机应用，2015, 35 (9): 2508-2512 (Chen Peng, Meng Chen, Wang Cheng. Simultaneous iterative hard thresholding for joint sparse recovery based on redundant dictionaries [J] .Journal of Computer Applications, 2015, 35 (9): 2508-2512)

[15] 唐文娟，赵生妹，郑宝玉．一种混合支撑集模型的联合半迭代硬阈值追踪重构算法 [J]. 信号处理，2015, 31 (8): 924-931. (Tang Wenjuan, Zhao Shengmei, Zheng Baoyu. Study on joint semi-iterative hard thresholding pursuit algorithm in mixed support-set model [J]. Signal Processing, 2015, 31 (8): 924-931)

[16] 牛亚坤，王振明，李陶深．一种压缩采样硬阈值追踪压缩感知重构算法 [J]. 计算机应用研究，2015, 32 (8): 2286-2288. (Niu Yakun, Yu Zhenming, Li Taoshen. Compressed sampling hard threshold pursuit algorithm with compressive sensing [J] , Application Research of Computers, 2015, 32 (8): 2286-2288)

[17] Jain P, Tewari A, Dhillon I S. Partial hard thresholding [J] , IEEE Trans on Information Theory, 2017, 63 (5) , 3029-3038

[18] Baraniuk R G, Foucart S, Needell D, *et al.* One-bit compressive sensing of dictionary-sparse signals [EB/OL].(2016-06-24). http://www.math.tamu.edu/~foucart/publi/Dict1Bit_v11.pdf.

[19] Hu Rui, Fu Yuli, Chen Zhen, *et al.* A lorentzian IHT for complex-valued sparse signal recovery [J]. Circuits Systems and Signal Processing, 2018, 37 (2): 862-872.

[20] Shalaby W A, Saad W, Shokair M, *et al.* Efficient parameters for compressed sensing recovery algorithms [J]. Wireless Personal Communications, 2017, 94 (3): 1715-1736.

[21] Wei Ke, Cai JianFeng, Chan T F, *et al.* Guarantees of Riemannian optimization for low rank matrix recovery [J]. SIAM Journal on Matrix Analysis and Applications, 2016, 37 (3): 1198-1222.

[22] Sabarinath G, Jose B R. A noise reduced hard thresholding method for noisy sparse reconstruction [J]. IOP Conference Series: Materials Science and Engineering, 2018, 396 (1): 1-8

[23] 孙涛．快速稀疏恢复算法研究 [D]，长沙：国防科技大学，2014 (Sun Tao. Fast sparse recovery algorithms [D]. Changsha: National University of Defense Technology, 2014)

[24] Foucart S. Hard thresholding pursuit: an algorithm for compressive sensing [J]. SIAM Journal on Numerical Analysis, 2011, 49, 2543-2563.

[25] Foucart S. Sparse recovery algorithms: sufficient conditions in terms of restricted isometry constants [C]//Approximation Theory XIII.New York: Springer, 2010,